UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/682,133	10/09/2003	Richard D. Dettinger	ROC920030237US1	1361

46797   7590   04/30/2008
IBM CORPORATION, INTELLECTUAL PROPERTY LAW
DEPT 917, BLDG. 006-1
3605 HIGHWAY 52 NORTH
ROCHESTER, MN 55901-7829

EXAMINER
PHAM, KHANH B

ART UNIT	PAPER NUMBER
2166	

MAIL DATE	DELIVERY MODE
04/30/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Application Number: 10/682,133
Filing Date: October 09, 2003
Appellant(s): DETTINGER ET AL.

Gero G. McClellan
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 2/12/2008 appealing from the Office action

mailed 9/12/2007.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

Scalzo, Bert , "Oracle DBA Guide to Data Warehousing and Star Schema", Prentice Hall, June 04, 2003, pages 1-10.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 102

1.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

2.   Claims **1-17, 26-41, 50-51** are rejected under 35 U.S.C. 102(a) as being anticipated by Bert Scalzo ("Oracle DBA Guide to Data Warehousing and Star Schema"), hereinafter "**Scalzo**".

As per claim 1, Scalzo teaches a method of managing execution of query operations in a data processing system comprising:

- "issuing, by a requesting entity, a request to perform a composite query operation defined by at least an initial query operation and a plurality of subsequent query operation to be executed against a data repository of the data processing system" at bottom of page 6, ("**upserts**" operation comprises update and insert operations) and page 7, (the "pro-C program" including "**while**" loop for executing multiple query operations on multiple records)

- "executing the initial query operation" at bottom of page 7, (executing

  "update_command");

- "determining an operation status of the initial query operation" at top of page 8,

  ("if update fails because record not found");

- "selecting one of the plurality of subsequent query operations based on the

  operation status; performing the selected subsequent query operation" at top of

  page 8, (Executing insert_command if update fails)

- "updating the operation status based on a result of the subsequent query

  operation" at page 7 (executing "fgets" to update operation status);

- "managing execution of any remaining subsequent query operations on the basis

  of the updated operation status" at page 7 (Checking the condition of the "while"

  loop to determine whether to continue execution or stop);

- "upon determining the composite query operation has completed, returning a

  result of the composite query operation to the request entity" at page 8

  (Termination of query execution if all records are processed).

As per claim 2, Scalzo teaches the method of claim 1, wherein "determining and

managing are performed by a composite query operations manager" at pages 7-8.

As per claim 3, Scalzo teaches the method of claim 1, wherein "the requesting

entity is an application and wherein the determining and managing are performed by a

composite query operations manager" at pages 7-8.

As per claim 4, Scalzo teaches the method of claim 1, wherein "the initial and the subsequent query are SQL statement" at pages 7-8.

As per claim 5, Scalzo teaches the method of claim 1, wherein "determining an operation status of the initial query operation comprises determining a number of items affected by the initial query operation" at pages 7-8.

As per claim 6, Scalzo teaches the method of claim 1, wherein "determining an operation status of the initial query operation comprises determining whether the initial query operation complete successfully" at page 8.

As per claim 7, Scalzo teaches the method of claim 1, wherein "determining an operation status of the initial query operation comprises determining, on the basis of a return code received upon completion of the initial query operation, whether the initial query operation completed successfully" at page 8.

As per claim 8, Scalzo teaches the method of claim 1, wherein "managing execution of the subsequent query operation comprises: executing the subsequent query operation only if the initial query operation did not complete successfully" at page 8.

As per claim 9, Scalzo teaches a method of managing execution of query

operations in a data processing system comprising:

- "issuing, by a requesting entity, a request to perform a composite query operation

  defined by at least an initial query operation and a plurality of subsequent query"

  at page 6, "upsert operation";

- "providing selection logic defining a next query operation of the composite query

  operation to e executed" at pages 7-8, the "Pro-C program";

- "providing a plurality of failure conditions for determining when a failure of the

  composite query operation occurs" at pages 7-8, the "Pro-C program";

- "managing, using a composite query operations manager, execution of the initial

  query operation and the plurality of subsequent query operation on the basis of

  the selection logic and the plurality of failure condition" at pages 7-8;

- "upon determining the composite query operation has completed, returning a

  result of the composite query operation to the request entity" at page 8

  (Termination of query execution if all records are processed).

As per claim 10, Scalzo teaches the method of claim 9, wherein at least one

failure condition of the plurality of failure conditions indicates the initial query operation

and an operation status of the initial query operation which indicates a failure of the

composite query operation" at pages 7-8.

As per claim 11, Scalzo teaches the method of claim 10, wherein "each other

failure condition of the plurality of failure conditions indicates a series of the initial query

operation and at least one of the plurality of subsequent query operations, and an operation status of the at least one of the plurality of subsequent operation which indicates a failure of the composite query operation" at pages 7-8.

As per claim 12, Scalzo teaches the method of claim 9, wherein "managing execution of the initial query operation and the plurality of subsequent query operations comprise a) "executing the initial query operation"; b) determining an operation status of the initial query operation"; c) determining, on the basis of the operation status and the plurality of failure condition, whether failure of the composite query operation occurred" d) "if no failure of the composite query operation occurred: i) determining the next operation to be executed from the plurality of subsequent query operations using the selection logic"; ii) "executing the next query operation"; iii) "determining, on the basis of the operation status of the next query operation"; iv) "determining, on the basis of the operation status and the plurality of failure conditions, whether failure of the composite query operation occurred"; v) "repeating step d) for at least one other of the plurality of subsequent query operations" at pages 7-8;

As per claim 13, Scalzo teaches the method of claim 12, wherein step c) and iv) comprises, if failure occurred: completing execution of the composite query operation, and returning a failure code as completion status of the composite query operation indicating a failure condition from the plurality of failure condition" at pages 7-8.

As per claim 14, Scalzo teaches the method of claim 12, wherein step b) and step iii) comprise determining whether the respective query operation complete successfully" at pages 7-8.

As per claim 15, Scalzo teaches the method of claim 12, wherein step b) and iii) comprise determining, on the basis of a return code received upon completion of the respective query operation, whether the respective query operation complete successfully" at pages 7-8.

As per claim 16, Scalzo teaches the method of claim 12, wherein step b) and iii) comprise determining whether the respective query operation completed successfully and, if the respective query operation completed successfully: completing executing the composite query operation" at pages 7-8.

As per claim 17, Scalzo teaches the method of claim 9, wherein "the initial and each subsequent query operation is an SQL statement" at pages 7-8.

Claims 26-41 and 50-51 recite a computer readable storage medium and a data processing system for performing similar method as in claims 1-17 and are therefore rejected by the same reasons.

(10) Response to Argument

1. Claims 1-17, 26-41, 50-51 are anticipated by Scalzo under 35 U.S.C §102(a).

Appellant's invention is directed to method for executing a **composite query**

operation defined by an **initial query** operation and a plurality of **subsequent query**

operations, wherein the executions of the subsequent query operations are based upon

an **operating status** of the initial query operation. For example, as shown at pages 24-

25, paragraphs [0071]-[0072] of the **Appellant Specification**, the execution of the

UPSERT composite query operation defined by an **INSERT** (i.e. "initial query

operation") and an **UP**DATE (i.e. "subsequent query operation"), which is implemented

using the algorithm listed in Table I reproduced below, wherein the UPDATE operation

is executed (line 004) only if the INSERT operation fails (line 003).

Table 1 (Appellant Specification page 24)

```
001    ImplemantationSchema1:
002    do Insert
003    if (failed) {
004            doUpdate
005            if (failed)      {
006                    return (failureCode)
007            }
008    }
```

Similarly, Scalzo teaches at page 6 of 10 the **UPSERT** operation, which is "an

UPDATE that INSERT when a record is not found". An implementation of the UPSERT

in form of C programming languages is shown at pages 7-8, which is a more detail

version of appellant's algorithm listed in Table I. A portion of Scalzo's UPSERT Pro-C-

program at pages 7-8 is reproduced below:

```
*******************************************************************************

/* Process data file records */
while (fget (rec, sizeof rec, fid) != NULL) {
.....
/* First - try to update existing record */
        EXEC SQL EXECUTE update_command
...
/* Second - if update fails because record not found, then insert record */
if (sqlca.sqlcode == 1403)
        EXEC SQL EXECUTE insert_command {
        ...
        }
else if (sqlca.sqlcode != 0)
...
}
*******************************************************************************
```

Response to Arguments Regarding claims 1-8, 26-32, and 50.

        Appellant argued that Scalzo does not teach a "method of managing execution of

query operation in a data processing system" that include "updating an operation status

based on a result of a subsequent query operation". The examiner respectfully

disagrees.

        As seen in the Pro-C program above, Scalzo utilizes a "**while**" loop to process

file records. The expression "fget (rec, sizeof rec, fid) != NULL" when executed results in

either "true" or "false", is operation condition which decides when the program will be

terminated to exit the while loop. The "fget" function takes three parameters: rec

(Record), sizeof rec (size of record), and fid (file ID) and returns a record read from the

inputted file. After each iteration of the while loop, the "fget" is executed and returns the

next record in the inputted file, the returned record is compared with "NULL" to

determine **current operation status**. At the end of the inputted file, "fget" will return

NULL, the operation status is changed to "FALSE" to indicate that last record has been

read and the program will exit the while loop. (Notes the notation "!=" means "not equal"

in C programming language, and NULL means "no thing").

Therefore, Appellant argument that the "Scalzo does not teach "updating an

operation status based on result of a subsequent query operation" is incorrect. Scalzo

updates the operation status from "TRUE" to "FALSE" to cause the program to exit the

while loop. Otherwise, if the operation status was not changed, the while loop will run

indefinitely and the Pro-C program will never be terminated.

Appelant further argued that "no query operation is performed by the fget call",

but admitted that "the while loop use the "fget" command to simply read in the next

record from an input file". Appellant's specification at paragraph [0005] defines "query"

as a set of commands that cause execution of **operation for retrieving data** from a

stored database. Therefore, fget is an query operation because it reads (i.e. "retrieves")

data records from an input data file.

Regarding claims 9-17, 33-41, and 51.

Appellant argued that Scalzo does not teach "managing, using a composite

query operation manager, execution of the initial query operation and the plurality of

subsequent query operations on the basis of the **selection logic** and the **plurality of failure condition**", the examiner respectfully disagrees.

Scalzo clearly teaches at pages 7-8 the step of executing an initial query operation (i.e. "update_command"), and based on the **plurality of failure conditions** (i.e, "if(sqlca.sqlcode == 1403)" and if (sqlca.sqlcode !=0)), performing the subsequent operation. Further, Scalzo teaches the UPDATE and INSERT operations are performed within a "WHILE" loop so that multiple operations are performed while the conditions in the WHILE loop are true. Scalzo therefore teaches at least an initial query operation and a plurality of subsequent query operations being executed based on the basis of selection logic and the plurality of failure conditions as claimed.

Appellant further argued that "the cited code does not disclose a "plurality of failure condition" at all", while completely ignored the two failure conditions pointed out by the examiner (i.e, "if(sqlca.sqlcode == 1403)" and if (sqlca.sqlcode !=0)). Scalzo uses the failure condition "if (sqlca.sqlcode == 1403)" twice in the program, the plain meaning of this code is, as shown in the comment above the codes, "if update fails because record not found, then insert record". Appellant also admitted that Scalzo's program has the structure "first try A, then try B", which clearly maps to the claimed limitation "selection logic". Scalzo therefore teaches the claimed limitation "execution of the initial query operation and the plurality of subsequent query operations on the basis of the **selection logic** and the **plurality of failure conditions**".

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the

Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Khanh B. Pham/

Primary Patent Examiner

Conferees:

/Hosain T Alam/

Supervisory Patent Examiner, Art Unit 2166

/Vincent Trans/

SPRE/QAS, TC 2100